Chapter 7
Data Rate Estimation

Andre Tkacenko and Marvin K. Simon

In an autonomous radio operation setting, one of the first parameters that we would like to estimate reliably would be the data rate of the received signal. Knowledge of this parameter is required to carry out maximum-likelihood (ML) detection [1] of other parameters, such as the carrier phase or modulation type. Although ML estimation of the data rate itself is statistically optimal, given that there is little to no a priori knowledge of the incoming signal, this approach is often difficult if not impossible to do in practice.

One mitigating factor for the autonomous radio under consideration is the fact that the data rates are assumed to come from a set of known values, such as the data rates used in the Electra radio (see [2] and Chapter 2). In particular, the data rates here are assumed to be related by integer powers of an integer base B. This assumption, as will soon be shown, allows us to estimate the *true* data rate based on estimates of the signal-to-noise ratio (SNR) computed for various *assumed* data rates. The method for estimating the SNR here is the split-symbol moments estimator (SSME) discussed in [3] and Chapter 6. This estimator is appealing in that the only parameter required for its operation is the assumed data rate. Hence, estimation of the data rate can be done *jointly* with that of the SNR.

Although this approach provides us with a way to estimate both the data rate and SNR together, it will be shown that it is sensitive to symbol-timing error or jitter. In fact, the presence of symbol-timing error can severely degrade the performance of this estimator, as shown in Chapter 6. To overcome this, a modification is proposed in which the jitter is quantized and estimated alongside the data rate and SNR. This approach, based on a so-called generalized likelihood ratio test (GLRT) [4], is robust in the presence of symbol-timing error and

can be used to jointly estimate the data rate, SNR, and symbol-timing error all at once. The estimates of the symbol-timing error obtained can then be used as coarse initial estimates for the data-transition tracking loop (DTTL), which is used later in the receiver to obtain a fine estimate of the timing jitter (see [5] and Chapter 10 for more details).

In Section 7.1, we review the received signal model assumptions and show how the SSME can be used to obtain an estimate of the data rate in the absence of symbol-timing error. This leads to an algorithm for estimating the data rate, which we present in Section 7.1.3. A slight modification to this algorithm that resembles a GLRT-type approach is presented in Section 7.1.4.

In Section 7.2, we investigate the effects of the presence of symbol-timing error on the data rate estimation algorithm. There, it is shown that the presence of a severe jitter can in fact cause the data rate estimator to unequivocally fail.

By quantizing the symbol-timing error, we show in Section 7.3 how to modify the algorithms in Sections 7.1.3 and 7.1.4 to account for the presence of symbol-timing error. There, an all-digital implementation of the SSME-based data rate estimation system is presented in Section 7.3.1. This leads to a joint data rate/SNR/symbol-timing error estimation technique that we describe in Section 7.3.2 and a GLRT-type modification to this method described in Section 7.3.3.

Simulation results for the joint data rate/SNR/symbol-timing error estimation techniques of Sections 7.3.2 and 7.3.3 are presented in Section 7.4. There, the strengths and weaknesses of each of the proposed techniques are revealed in terms of probability of data rate misclassification, SNR estimation error, and jitter estimation error.

7.1 Data Rate Estimation Based on the Mean of the SSME SNR Estimator

7.1.1 Signal Model and Assumptions

The baseband signal received at the autonomous radio is assumed to consist of a constant amplitude digital data stream corrupted only by artifacts due to the conversion from intermediate frequency (IF) to baseband as well as to additive noise. Mathematically, the received signal $\tilde{r}(t)$ is assumed to have the following form in the complex baseband representation:

$$\tilde{r}(t) = A\left(\sum_{k=-\infty}^{\infty} d_k p\big(t-(k+\varepsilon)T\big)\right) e^{j(\omega_r t+\theta_c)} + \tilde{n}(t) \tag{7-1}$$

Here, we have the following:

$A =$ signal amplitude

$d_k =$ kth data symbol, typically assumed to be an M-PSK symbol [1]

$p(t) =$ data pulse shape, typically either a non-return to zero (NRZ) or a Manchester pulse [1]

$T =$ symbol period of the data

$\varepsilon =$ symbol-timing error (jitter), assumed to be uniform over the interval $[0, 1)$

$\omega_r =$ residual frequency offset after demodulation and frequency correction

$\theta_c =$ carrier phase, assumed to be uniform over the interval $[0, 2\pi)$

$\tilde{n}(t) =$ complex additive white Gaussian noise (AWGN) whose real and imaginary parts are uncorrelated, zero-mean processes with two-sided power spectral density (psd)[1] $N_0/2$

Prior to estimating parameters such as the carrier phase θ_c or the frequency offset ω_r, we would like to estimate the data rate given by $\mathcal{R} \triangleq 1/T$. As with the Electra radio (see Chapter 2), we assume that the set of possible data rates $\{\mathcal{R}\}$ comes from a *known* finite set of values of the form

$$\mathcal{R} = B^{\ell}\mathcal{R}_b, \quad 0 \le \ell \le \ell_{\max} \tag{7-2}$$

where B, ℓ, and $\ell_{\max}$ are nonnegative integers and $\mathcal{R}_b \triangleq 1/T_b$ is the *basic* (or lowest) data rate. In other words, every possible data rate is a *base power* of the lowest basic rate. Here B is called the *rate base*, whereas ℓ is referred to as the *rate power*. We denote the maximum rate power by $\ell_{\max}$, and so the number of possible data rates is given by $(\ell_{\max} + 1)$, as can be seen from Eq. (7-2). For the Electra radio [2], we have

$B = 2$

$\ell_{\max} = 12$

$\mathcal{R}_b = 1$ ksymbol/s

With regard to estimating the data rate of the signal $\tilde{r}(t)$ from Eq. (7-1), it is assumed that we know both the rate base B and the basic data rate $\mathcal{R}_b$.

[1] The two-sided psd of the real and imaginary parts is defined to be $N_0/2$ here to set the two-sided psd of the *complex* baseband noise process $n(t)$ at N_0. This is a standard notational convention for a complex baseband AWGN process [1]. It should be noted here that the parameter N_0 differs from that used for the passband process $n(t)$ appearing in other chapters of the book by a factor of two.

Hence, from Eq. (7-2), the only ambiguity of the data rate that exists is the rate power ℓ. This greatly simplifies the data rate estimation problem, since ℓ only varies over a finite set of *known* integers. In what follows, we will assume that the symbol-timing error ε is zero. The case for which $\varepsilon \neq 0$ will be considered in Section 7.2.

7.1.2 Relation of the SSME SNR Estimator to Data Rate Estimation

A block diagram of the SSME system for estimating the SNR of the signal $\tilde{r}(t)$ from Eq. (7-1) is shown in Fig. 7-1 for the case of a rectangular NRZ pulse shape. (For different pulse shapes, the only thing that needs to be changed is that the half-symbol integrate-and-dump (I&D) circuits need to be replaced with half-symbol matched filters [1].) Here, T_s denotes the assumed symbol period of the system (i.e., the sample period); N_s denotes the number of observations; and ω_{sy}, $\hat{h}^+$, and $\hat{h}^-$ denote frequency and phase compensation factors as described in Chapter 6.

From Chapter 6, it is known that if the assumed data rate $\mathcal{R}_s \triangleq 1/T_s$ and N_s satisfy

$$\mathcal{R}_s = L\mathcal{R}, \quad N_s = LN$$

for some positive integers L and N, then the mean of the SNR estimate $\hat{R}_\ell$ is given as follows:

$$E\left[\hat{R}_\ell\right] = \frac{RN+1}{LN-1} = \frac{\dfrac{R}{L}+\dfrac{1}{LN}}{1-\dfrac{1}{LN}} = \frac{R}{L} + \frac{1}{LN}\left(\frac{R}{L}+1\right) + O\left(\frac{1}{N^2}\right)$$

where R is the true SNR given by $R = (A^2T)/N_0$. For large N, this simplifies within $O(1/N)$ to become

$$E\left[\hat{R}_\ell\right] \cong \frac{R}{L} \tag{7-3}$$

In other words, if the assumed data rate $\mathcal{R}_s$ is an integer multiple L of the true data rate $\mathcal{R}$, then the SSME still works as before, but it formulates an estimate of the reduced SNR R/L when the number of observations is large enough. As we shall soon see, it is this property that will allow us to use the SSME system to estimate the data rate.

To see how the SSME can be used to estimate the data rate, suppose first that the SSME operates at the highest possible rate, which is simply $\mathcal{R}_s = B^{\ell_{\max}}\mathcal{R}_b$

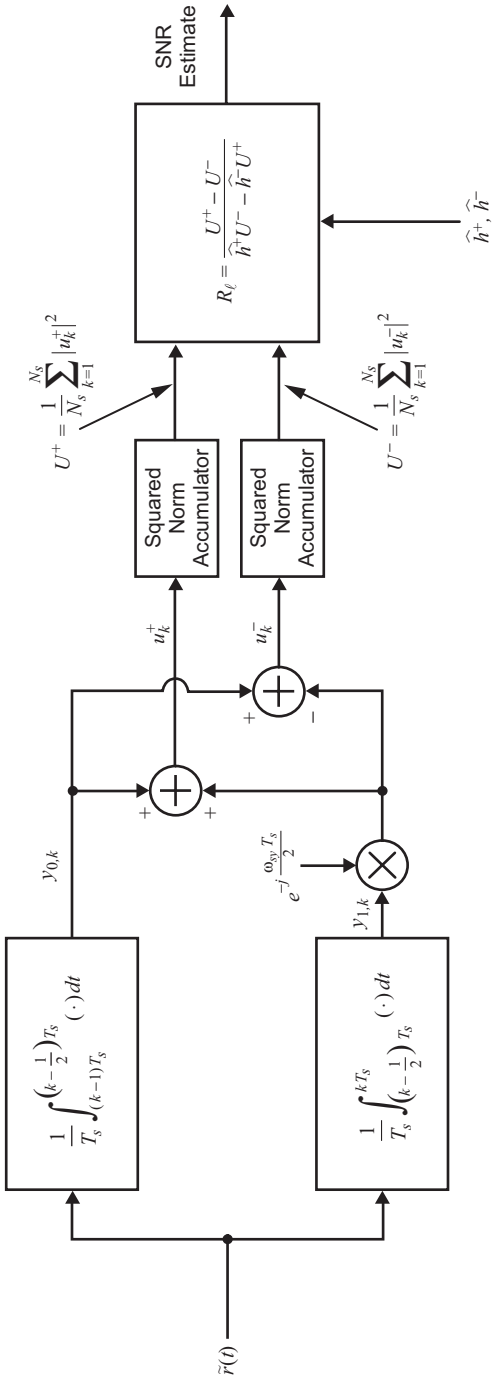

Fig. 7-1. Split-symbol moments estimator (SSME) for a rectangular NRZ pulse shape.

from Eq. (7-2). As $\mathcal{R} = B^{\ell}\mathcal{R}_b$, we have $\mathcal{R}_s = L\mathcal{R}$, where $L = B^{\ell_{\max}-\ell}$. Then, from Eq. (7-3), we have,

$$E\left[\hat{R}_0\right] = \frac{R}{B^{\ell_{\max}-\ell}} + O\left(\frac{1}{N}\right)$$

If the SSME is operated at the next lower rate (i.e., $\mathcal{R}_s = B^{\ell_{\max}-1}\mathcal{R}_b$), then we have $L = B^{\ell_{\max}-\ell-1}$, and so from Eq. (7-3) we have

$$E\left[\hat{R}_1\right] = \frac{R}{B^{\ell_{\max}-\ell-1}} + O\left(\frac{1}{N}\right) = BE\left[\hat{R}_0\right] + O\left(\frac{1}{N}\right)$$

In other words, lowering the rate by one step *increases* the mean of the SNR estimate by a factor of B.

If we continue to run the SSME, lowering the assumed data rate $\mathcal{R}_s$ by a factor of B at each run, then on the $(\ell_{\max} - \ell)$th run, we will obtain an SNR estimate based on the *true* data rate $\mathcal{R}$, in which case we have

$$E\left[\hat{R}_{\ell_{\max}-\ell}\right] = R + O\left(\frac{1}{N}\right) = B^{\ell_{\max}-\ell}E\left[\hat{R}_0\right] + O\left(\frac{1}{N}\right)$$

Note that up to this point we have

$$E\left[\hat{R}_i\right] = B^i E\left[\hat{R}_0\right] + O\left(\frac{1}{N}\right) \tag{7-4}$$

In other words, the mean of the SNR estimate *monotonically increases* by a factor of B each time the rate is lowered until the true data rate (and hence the true SNR) is reached.

If the assumed data rate is lowered one more step so that $\mathcal{R}_s = B^{\ell-1}\mathcal{R}_b = (1/B)\mathcal{R}$, then the SSME will attempt to create an SNR estimate based on B successive data symbols. This will severely degrade the performance of the estimator since the data symbols fluctuate randomly. To see this, consider the case where $B = 2$ and the data come from a binary phase-shift keying (BPSK) constellation [1]. In this case, the signal portion of the I&D outputs $y_{0,k}$ and $y_{1,k}$ can either constructively or destructively interfere depending on whether adjacent data symbols are the same or different, respectively. This is illustrated in Fig. 7-2.

When two adjacent data symbols are the same, as in Fig. 7-2(a), we will get a valid contribution to the SNR estimate, since $|u_k^+|^2$ from Fig. 7-1 will be an approximate measure of the signal power plus the noise power, whereas $|u_k^-|^2$

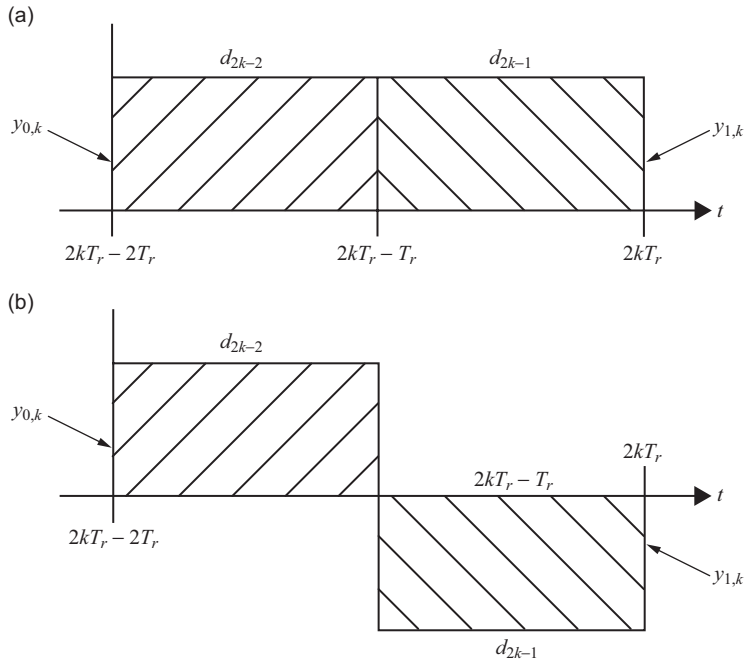

Fig. 7-2. Signal component of the I&D outputs $y_{0,k}$ and $y_{1,k}$ when the SSME assumed data rate is half of the true data rate for the case of (a) identical and (b) different adjacent data symbols.

will be a measure of the noise power. However, when two adjacent data symbols are different, as in Fig. 7-2(b), the opposite scenario takes place, i.e., $|u_k^+|^2$ becomes a measure of the noise power whereas $|u_k^-|^2$ becomes a measure of the signal-plus-noise power. This will result in a severely degraded estimate of the SNR since half of the time adjacent data symbols will be the same and half of the time they will be different. (The reason for this is that the data sequence is assumed to come from an independent, identically distributed (iid) source [1].) This degradation may even lead to negative estimates of the SNR which are clearly absurd.

For the purpose of data rate estimation, this degradation can be used to indicate that the assumed data rate of the SSME system was lowered excessively by one step. The elegance of this method of estimating the data rate is the rapid degradation that is expected once the assumed data rate has been lowered beyond the true data rate. Recall from Eq. (7-4) that up until the true data rate is reached, the mean of the SNR estimate will *increase* by a factor of B until the true SNR is reached. Once the assumed data rate is lowered by one more step, however, the mean of the SNR estimate will *decrease* significantly. Hence, the

SSME provides us with a way to estimate the data rate via a sharp transition in the estimate of the SNR.

An algorithm to estimate the data rate based on this phenomenon is presented below.

7.1.3 SSME Data Rate Estimation Algorithm

(1) Assume that the data rate is the maximum rate, i.e., set $\mathcal{R}_s = B^{\ell_{\max}}\mathcal{R}_b$. Run the SSME and compute an estimate of the mean of the SNR and call it $\hat{\mu}_{\hat{R}_0}$. Set $i = 1$.

(2) Lower the SSME data rate by a factor of B, i.e., set $\mathcal{R}_{s,\text{new}} = (1/B)\mathcal{R}_{s,\text{old}}$. Compute an estimate of the SNR mean and call it $\hat{\mu}_{\hat{R}_i}$.

(3) If $\hat{\mu}_{\hat{R}_i} \geq \hat{\mu}_{\hat{R}_{i-1}}$, then increment i by 1 and go to Step (2). Otherwise stop and estimate the SNR to be $\hat{\mu}_{\hat{R}} = \hat{\mu}_{\hat{R}_{i-1}}$ and the data rate to be $\hat{\mathcal{R}} = B^{\ell_{\max}-(i-1)}\mathcal{R}_b$.

In practice, the estimate of the SNR mean $\hat{\mu}_{\hat{R}_i}$ is computed as an *ensemble average* of observed SNR estimates $\hat{R}_i$ calculated over several *blocks* of the received signal. If a large enough ensemble of blocks is used, then we will have $\hat{\mu}_{\hat{R}_i} \approx E[\hat{R}_i]$ as desired.

It should be noted that this algorithm terminates as soon as $\hat{\mu}_{\hat{R}_i} < \hat{\mu}_{\hat{R}_{i-1}}$. In other words, the assumed data rate of the SSME is lowered only until the condition $\hat{\mu}_{\hat{R}_i} \geq \hat{\mu}_{\hat{R}_{i-1}}$ is not satisfied. Although this approach works in theory assuming that the number of observations is large enough, in practice this can often lead to a premature termination of the algorithm depending on the value of the *variance* of the SSME SNR estimate. (See [3] and Chapter 6 for more details.) For cases where the SNR is low, such as often occurs in the Deep Space Network (DSN), this can lead to a perturbation in the calculation of the mean of the SNR such that the condition $\hat{\mu}_{\hat{R}_i} < \hat{\mu}_{\hat{R}_{i-1}}$ will occur before it should, causing the algorithm to halt prematurely.

Since we expect the largest SNR to occur when the assumed data rate is equal to the true data rate, one alternative to this algorithm is to run the SSME for all data rates and estimate the data rate as the one yielding the largest SNR mean. This forms the basis for the GLRT-type data rate estimation algorithm presented below.

7.1.4 GLRT-Type SSME Data Rate Estimation Algorithm

(1) Run the SSME for all data rates and (as before) let $\hat{\mu}_{\hat{R}_i}$ denote the estimate of the mean of the SNR for the ith *largest* data rate.

(2) Define the optimal index i_{opt} to be $i_{\text{opt}} \triangleq \underset{0 \le i \le \ell_{max}}{\text{argmax}}\ \hat{\mu}_{\hat{R}_i}$. Then, estimate the true SNR and data rate as follows:

$$\hat{\mu}_{\hat{R}} = \hat{\mu}_{\hat{R}_{i_{\text{opt}}}}$$

$$\hat{\mathcal{R}} = B^{\ell_{\max} - i_{\text{opt}}} \mathcal{R}_b$$

For a traditional GLRT estimator, the conditional-likelihood function (CLF) [4] of the observables is *maximized* over the unknown parameters, as opposed to being *averaged* over them as is done in ML estimation. In that sense, this algorithm is a GLRT-like approach in that the SNR is chosen to be the maximum value obtained over the *unknown parameter* of the data rate. The data rate, in turn, is estimated as the rate that yields the largest SNR mean.

As will be shown in Section 7.4 through simulations, the GLRT-type data rate estimation algorithm outperforms the algorithm of Section 7.1.3 for low SNR when the true data rate is the lowest data rate. The reason for this is that this algorithm calculates an estimate of the SNR for *all* rates and doesn't prematurely terminate as the previous algorithm may do.

Prior to showing simulation results for these algorithms, we first investigate the effects of the presence of symbol-timing error on estimating the data rate. There, we show that these effects can seriously adversely affect the performance of the above proposed data rate estimation algorithms. In Sections 7.3.2 and 7.3.3, we present modifications to the algorithms of Sections 7.1.3 and 7.1.4, respectively, which account for the presence of symbol-timing error.

7.2 Effects of Symbol-Timing Error on Estimating the Data Rate

In the previous section, we assumed that the symbol-timing error or jitter ε was zero. From Chapter 6, it is known that the presence of jitter will have the effect of degrading the estimate of the SNR of the SSME. Heuristically speaking, the reason for this is that the half-symbol I&D outputs will contain the contributions of two adjacent data symbols. As the data symbols are iid, the signal components of the I&D outputs will be degraded similarly to the way in which they were degraded in Section 7.1.2 when the assumed data rate was lower than

the true data rate. This effect becomes more pronounced as ε reaches its worst case value of $1/2$.

To mitigate the effects of the presence of a nonzero ε, the approach suggested in Chapter 6 was to increase the data rate of the SSME system by a factor of L. By doing so, the vast majority of the half-symbol I&D outputs contain contributions due to only one data symbol, as desired. The effects due to those containing contributions from two adjacent data symbols become negligible, and so the oversampled estimator is then robust to the presence of jitter.

This principle of oversampling is used in the data rate estimation algorithms of Section 7.1. There, the oversampling factor is reduced at each stage until the largest SNR mean is obtained. The problem with these algorithms in the presence of symbol-timing error is that the SNR will appear to be degraded once the assumed data rate is lowered to the *true* data rate and not just afterward. In other words, for non-negligible values of the jitter, the largest SNR mean obtained will not occur when the SSME is operating at the true data rate, and so the data rate will be estimated erroneously. Furthermore, the SNR estimated will be far from its true value (approximately off by a factor of a power of B), since the data rate was incorrectly classified.

As an example to illustrate the adverse effects of symbol-timing error on the estimation of the data rate, consider the special case where $\varepsilon = 1/4$ and we have BPSK data as in the example in Section 7.1.2. Suppose that the system data rate of the SSME is equal to that of the true data rate. Then depending upon whether adjacent data symbols are the same or different, the signal portions of the I&D outputs $y_{0,k}$ and $y_{1,k}$ will be unaltered or degraded, respectively, as shown in Fig. 7-3.

Just as with the example considered in Section 7.1.2, when two adjacent data symbols are the same as in Fig. 7-3(a), we will obtain a valid contribution to the SNR estimate, since $y_{0,k}$ and $y_{1,k}$ will contain the same signal component support and polarity. However, when the adjacent data symbols are different as in Fig. 7-3(b), then we will have $y_{0,k} = 0$, which will severely degrade the SNR estimate. The reason for this is that, in this case, neither $|u_k^+|^2$ will be a good measure of the signal-plus-noise powers nor will $|u_k^-|^2$ be a good measure of the noise power. Instead, $|u_k^+|^2$ and $|u_k^-|^2$ will be measures of essentially the same quantity, namely a combination of *half* of the signal power together with the *full* noise power. This will result in a poor estimate of the SNR.

7.2.1 Accounting for the Symbol-Timing Error

To account for the presence of symbol-timing error, typically a data-transition tracking loop (DTTL) is used (see [5] and Chapter 10 for more details). However, a typical DTTL requires knowledge of *both* the carrier phase and data rate in order to operate properly. Thus, it appears as though there is a

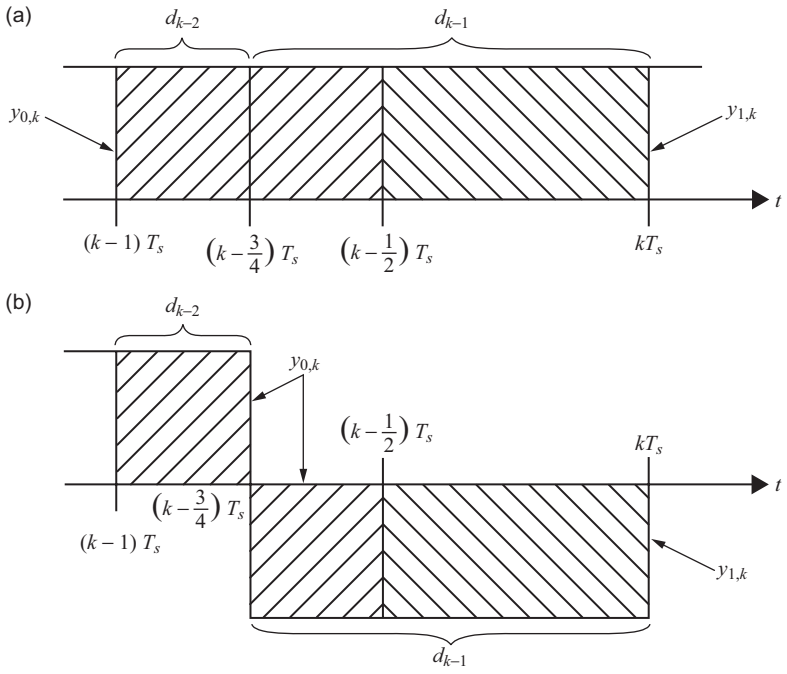

Fig. 7-3. Signal component of the I&D outputs $y_{0,k}$ and $y_{1,k}$ when the symbol timing error is $\varepsilon = 1/4$ for the case of (a) identical and (b) different adjacent data symbols.

dilemma. The data rate estimation algorithms of Sections 7.1.3 and 7.1.4 cannot reliably estimate the data rate (or the SNR for that matter) in the presence of symbol-timing error, and the symbol-timing error cannot be estimated without knowledge of the data rate (as well as the carrier phase).

To overcome this dilemma, we will exploit the fact that on average the presence of symbol-timing error only has a deleterious effect on the SNR estimate as shown in Chapter 6. The approach that will be taken here is to *quantize* the assumed symbol-timing error to a finite number of levels. Then, for each data rate, the SSME is run for each quantized jitter value. The SNR then is estimated to be the largest SNR obtained while the jitter is estimated as the value that yielded the largest SNR mean. In this way, not only do we obtain an improved estimate of the SNR for each assumed data rate, but we also obtain a coarse estimate of the symbol-timing error itself.

Hence, we generalize the data rate estimation algorithms of Sections 7.1.3 and 7.1.4 to jointly estimate the data rate, SNR, and symbol-timing error. Even with a coarse quantization of the symbol-timing error, this leads to a rather robust estimation of the data rate in the presence of jitter, as will be shown through simulations in Section 7.4. Once a reliable estimate of the data rate

has been made, the DTTL then can be used to obtain a finer estimate of the symbol-timing error. Furthermore, the coarse estimate of the jitter can be used as an initial condition for the DTTL which may reduce the computation time required for convergence.

It should be noted that this approach is different from the one suggested in Chapter 10, Section 10.8, in which oversampling is used to obtain a coarse estimate of the symbol-timing error. There, the data rate is assumed to be known, and the jitter is estimated by exploiting the fact that the presence of symbol-timing error becomes less noticeable as the oversampling ratio $L \to \infty$. This approach doesn't necessitate a modification to the SSME structure shown in Fig. 7-1, whereas the method suggested here does, as we show in the next section.

7.3 Quantization of the Symbol-Timing Error

As the data rate of the received signal is not known a priori, at the receiver, we are at liberty to independently quantize only the symbol-timing error corresponding to one specific data rate. The reason for this is that, by quantizing the jitter corresponding to one data rate, the quantized jitters for the remaining rates are automatically determined. In order to ensure that we have, say, at least $N_{\hat{\varepsilon},b}$ quantization levels for all rates, we must quantize the symbol-timing error corresponding to the *highest rate* by at least $N_{\hat{\varepsilon},b}$ levels. The reason for this is that if the highest data rate symbol-timing error is quantized to $N_{\hat{\varepsilon},b}$ steps, then the number of jitter steps at the next lowest rate will be $BN_{\hat{\varepsilon},b}$. By inductive argument, the number of quantization levels of the symbol-timing error at the kth lowest data rate will be $B^k N_{\hat{\varepsilon},b}$.

Following this logic, at the receiver, the symbol-timing error ε will be assumed to be *uniformly* quantized to $\hat{\varepsilon} = n/N_{\hat{\varepsilon},s}$ for some $0 \leq n \leq N_{\hat{\varepsilon},s} - 1$, where we have

$$N_{\hat{\varepsilon},s} = B^{\ell_{\max} - \ell_s} N_{\hat{\varepsilon},b} \tag{7-5}$$

Here, $N_{\hat{\varepsilon},b}$ denotes the *basic* number of jitter quantization steps (i.e., the number of steps at the *highest* data rate), whereas $N_{\hat{\varepsilon},s}$ denotes the *system* number of jitter quantization steps (i.e., when the assumed data rate power is ℓ_s).

Since the number of quantization steps increases as the assumed data rate decreases, it is tempting to think that we will always obtain a better estimate of the data rate, SNR, and symbol-timing error for lower true data rates than for higher rates. However, this is offset by the fact that, for a fixed observation time interval, we will obtain a larger number of observations for higher true data rates than for lower ones. Hence, we have an

implicit trade-off between the number of signal observations and the number of jitter quantization levels for each true data rate.

One of the advantages of uniformly quantizing the symbol-timing error to $N_{\hat{\varepsilon},s}$ steps as in Eq. (7-5) is that it leads to an efficient *all-digital* implementation of the SSME system, as we now proceed to show.

7.3.1 All-Digital Implementation of the SSME-Based Data Rate Estimator

Suppose that, prior to processing the received signal $\tilde{r}(t)$ from Eq. (7-1) through the SSME, it is *finely* integrated and sampled to obtain the discrete-time signal y_m using the system of Fig. 7-4. Here, $T_{\min}$ is the *time resolution period* given to be

$$T_{\min} \triangleq \frac{T_b}{B^{\ell_{\max}} N_{\hat{\varepsilon},b}} = \frac{1}{N_{\hat{\varepsilon},b}\,(B^{\ell_{\max}} \mathcal{R}_b)}$$

Note that $T_{\min}$ is $N_{\hat{\varepsilon},b}$ times *smaller* than the *shortest* possible data symbol interval. Equivalently, $1/T_{\max}$ is $N_{\hat{\varepsilon},b}$ times larger than the highest possible data rate, as can be seen from Eq. (7-2).

To generalize the SSME structure of Fig. 7-1 to account for the quantized symbol-timing error, it is also necessary to generalize it to account for computing an ensemble average of the observed SNRs. Recall from Section 7.1.3 that an ensemble average of the observed SNRs is required in order to estimate the mean of the SNR of the SSME system. To do this, we partition the discrete-time signal y_m into blocks over which the SNR is to be computed. For each block, the SSME computes an estimate of the SNR, and then an ensemble average of the SNR is computed over the blocks.

Let N_{obs} denote the basic number of symbols to observe per block to obtain an SNR estimate (i.e., the number of symbols to observe per block at the *lowest*

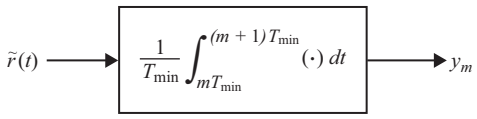

Fig. 7-4. System to finely integrate and sample the continuous-time signal $\tilde{r}(t)$ to obtain the high rate discrete-time signal y_m.

rate) and let N_b denote the number of blocks over which to compute an ensemble average of the SNR. Then, an all-digital implementation of the SSME system of Fig. 7-1 that accounts for the quantized symbol-timing error and ensemble averaging of the observed SNRs is shown in Fig. 7-5.

There are several things to note regarding the structure shown in Fig. 7-5. First, notice that the I&D half symbol integrators from Fig. 7-1 can be replaced with discrete summations, which is analogous to the sampled version of the SNR estimator discussed in Section 6.1.1.[2] Furthermore, note that all of the signals starting from the half-symbol integrator outputs are indexed with a semicolon followed by n. This notation was chosen here to reflect the fact that these quantities are parameterized by the quantized symbol-timing error $\hat{\varepsilon} = n/N_{\hat{\varepsilon},s}$, where the parameter n is an integer in the range

$$0 \leq n \leq N_{\hat{\varepsilon},s} - 1 \Longleftrightarrow 0 \leq n \leq B^{\ell_{\max}-\ell_s} N_{\hat{\varepsilon},b} - 1$$

Finally, note that to form a single SNR estimate, a total of $B^{\ell_s} N_{\text{obs}}$ samples are squared and accumulated. This was chosen as such here to keep the total observation time interval or epoch per block *fixed*.

By tracing the temporal indices m, k, and q from Fig. 7-5 backwards, it can be seen that, in order to have $0 \leq q \leq N_b - 1$ as desired, we need

$$0 \leq k \leq \left(B^{\ell_s} N_{\text{obs}}\right) N_b - 1$$

From this, it is clear that k must vary over an interval of N_b blocks each of size $B^{\ell_s} N_{\text{obs}}$, as desired and expected. Finally, from this range of the index k, in order to be able to accommodate all $N_{\hat{\varepsilon},s}$ values of the parameter n, it can be shown that the time index m should vary over the interval

$$0 \leq m \leq \left(N_{\hat{\varepsilon},b} B^{\ell_{\max}}\right) \left(N_{\text{obs}} N_b + 1\right) - 2 \tag{7-6}$$

To incorporate the estimation of the quantized symbol-timing error, the only required modification to the data rate estimation algorithms of Sections 7.1.3 and 7.1.4 is that the SNR estimate $\hat{\mu}_{\hat{R};n}$ must be calculated for each n. For a

[2] If the pulse shape used has some other piecewise constant shape such as the Manchester pulse, then the half-symbol integrators should be replaced with equivalent half-symbol digital matched filters [1]. In this way, the pulse shape also can be initially classified using an appropriate SSME for each possible pulse shape. This coarse pulse shape estimate can be used to judge the confidence of the estimation process when compared with the statistically optimal ML estimation approach of [6] and Chapter 5.

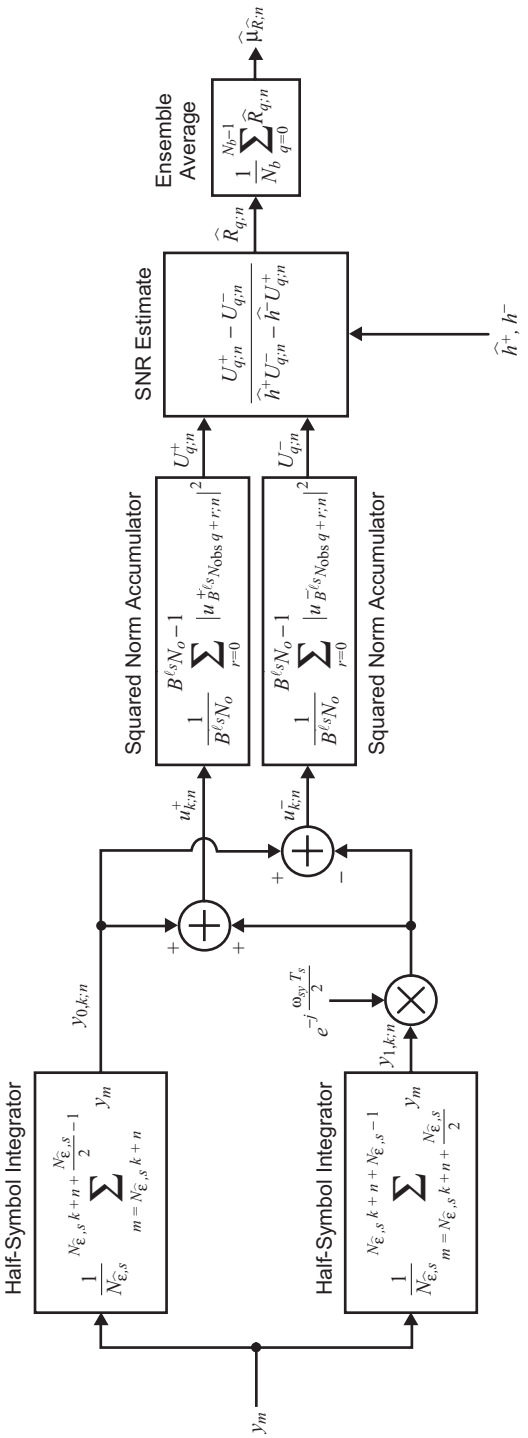

Fig. 7-5. Digital implementation of the SSME-based data rate estimation system that accounts for a quantized symbol-timing error and ensemble averaging of the observed SNRs.

fixed assumed data rate, the SNR is chosen to be the largest value of $\hat{\mu}_{\hat{R};n}$ while n is chosen to be the maximizing value of $\hat{\mu}_{\hat{R};n}$. This modification is described in the following algorithms.

7.3.2 SSME Data Rate/SNR/Symbol-Timing Error Estimation Algorithm

(1) Calculate the sequence y_m from Fig. 7-4 over the range of values given in Eq. (7-6).

(2) Run the SSME of Fig. 7-5 at the highest data rate $\mathcal{R}_s = B^{\ell_{\max}}\mathcal{R}_b$. Calculate $\hat{\mu}_{\hat{R};n}$ for all n and define $n_0 \stackrel{\triangle}{=} \underset{n}{\operatorname{argmax}}\, \hat{\mu}_{\hat{R};n}$ and $\hat{\mu}_{\hat{R}_0} \stackrel{\triangle}{=} \hat{\mu}_{\hat{R};n_0}$. Set $i = 1$.

(3) Lower the assumed data rate by one step, i.e., set $\mathcal{R}_{s,\text{new}} = (1/B)\mathcal{R}_{s,\text{old}}$, and run the SSME. Calculate $\hat{\mu}_{\hat{R};n}$ for all n and define $n_i \stackrel{\triangle}{=} \underset{n}{\operatorname{argmax}}\, \hat{\mu}_{\hat{R};n}$ and $\hat{\mu}_{\hat{R}_i} \stackrel{\triangle}{=} \hat{\mu}_{\hat{R};n_i}$.

(4) If $\hat{\mu}_{\hat{R}_i} \geq \hat{\mu}_{\hat{R}_{i-1}}$, increment i by 1 and go to Step (3). Otherwise, estimate the data rate, SNR, and symbol-timing error as follows:

$$\hat{\mathcal{R}} = B^{\ell_{\max}-(i-1)}\mathcal{R}_b$$

$$\hat{\mu}_{\hat{R}} = \hat{\mu}_{\hat{R}_{i-1}}$$

$$\hat{\varepsilon} = \frac{n_{i-1}}{B^{\ell_{\max}-(i-1)}N_{\hat{\varepsilon},b} - 1}$$

As mentioned above, for each assumed data rate, the SSME is run for each value of the quantized symbol-timing error. The SNR and jitter for that data rate then are estimated to be the largest SNR and the jitter value leading to this maximum SNR. Like the algorithm of Section 7.1.3, this data rate estimation technique halts as soon as the condition $\hat{\mu}_{\hat{R}_i} \geq \hat{\mu}_{\hat{R}_{i-1}}$ is not satisfied. This may lead to a premature termination of the algorithm as described in Section 7.1.3. To prevent a premature halting of the algorithm, a GLRT-type modification to the algorithm of Section 7.3.2 is proposed, similar to what was proposed in Section 7.1.4.

7.3.3 GLRT-Type SSME Data Rate/SNR/Symbol-Timing Error Estimation Algorithm

(1) Calculate the sequence y_m from Fig. 7-4 over the range of values given in Eq. (7-6).

(2) Run the SSME for all data rates and all possible quantized symbol-timing error values. Let $\hat{\mu}_{\hat{R};(i,n)}$ denote the estimate of the mean of the SNR for the ith largest data rate with quantized jitter value n. (Here we have $0 \leq i \leq \ell_{\max} - 1$ and $0 \leq n \leq B^{\ell_{\max}-i} N_{\hat{\varepsilon},b} - 1$.)

(3) Let i_{opt} and n_{opt} denote the indices for which $\hat{\mu}_{\hat{R};(i,n)}$ reaches its maximum value, i.e., i_{opt} and n_{opt} are such that $\hat{\mu}_{\hat{R};(i_{\text{opt}},n_{\text{opt}})} = \max_{i,n} \hat{\mu}_{\hat{R};(i,n)}$. Then, estimate the data rate, SNR, and symbol-timing error as follows:

$$\hat{\mathcal{R}} = B^{\ell_{\max}-i_{\text{opt}}} \mathcal{R}_b$$

$$\hat{\mu}_{\hat{R}} = \hat{\mu}_{\hat{R};(i_{\text{opt}},n_{\text{opt}})}$$

$$\hat{\varepsilon} = \frac{n_{\text{opt}}}{B^{\ell_{\max}-i_{\text{opt}}} N_{\hat{\varepsilon},b} - 1}$$

This GLRT-type estimation algorithm is based on the principle that the true data rate and symbol-timing error should yield the largest value of the mean of the SNR. Incorrect values of these quantities, on the other hand, should lead to a degraded estimate of the SNR mean. As opposed to the previous algorithm, which lowers the assumed data rate until the SNR decreases, this algorithm computes the SNR for all data rates and all jitter values. The advantage to this is that it can prevent the algorithm from prematurely terminating, which easily can happen when the true SNR is low. This is especially the case when the true data rate is low, as we show through simulations in the next section.

7.4 Simulation Results for the SSME-Based Estimation Algorithms

In order to properly evaluate the performance of the estimation algorithms of Sections 7.3.2 and 7.3.3, we must consider different metrics for each of the parameters that we wish to estimate. Prior to presenting simulation results, we introduce these metrics and justify their usage here.

7.4.1 Performance Metrics Used for Evaluating the Estimation Algorithms

For all of the following measures used, we assume that the estimation algorithms have each been run for a total of N_t trials. Parameters estimated at the nth trial (where $0 \leq n \leq N_t - 1$ for simplicity) are denoted with a superscript surrounded by parentheses. For example, the data rate estimated at the nth trial is denoted as $\hat{\mathcal{R}}^{(n)}$.

7.4.1.1. Probability of Data Rate Misclassification. In order to assess the performance of the algorithms with respect to estimating the data rate, one valid measure of performance is the empirical probability of data rate misclassification, which is defined below:

$$P_m \triangleq \frac{1}{N_t} \sum_{n=0}^{N_t-1} \mathcal{I}\left(\hat{\mathcal{R}}^{(n)} \neq \mathcal{R}\right) \tag{7-7}$$

where $\mathcal{I}(X)$ is an indicator function that is unity if the event X is true and zero if X is false. From Eq. (7-7), it is clear that $0 \leq P_m \leq 1$ and that P_m is a linear measure of the number of times each algorithm fails to estimate the data rate correctly.

7.4.1.2. Mean-Squared SNR Decibel Estimation Error. To properly gauge the performance of the estimation algorithms with respect to estimating the SNR, we seek a metric that penalizes the error between the estimated and true SNRs based on the value of the true SNR. In particular, small differences in SNR should be penalized more so if the true SNR is small than if it is large. For example, if the true SNR is 1 and the SNR is estimated to be 0.7, then it is reasonable to penalize this error more so than if the true SNR were 100 and the estimated SNR were 97.

One metric that penalizes the error in the SNR in such a way is the mean-squared error between the estimated and true SNRs in decibels (dB). This measure is the mean-squared SNR dB estimation error and is given below as follows:

$$\xi_R \triangleq \frac{1}{N_t} \sum_{n=0}^{N_t-1} \left|\hat{\mu}_{\hat{R}}^{(n)}\,(\mathrm{dB}) - R\,(\mathrm{dB})\right|^2 = \frac{1}{N_t} \sum_{n=0}^{N_t-1} \left|10\log_{10}\left(\frac{\hat{\mu}_{\hat{R}}^{(n)}}{R}\right)\right|^2 \tag{7-8}$$

From Eq. (7-8), it is clear that for low true SNR a deviation from the true SNR is penalized more so than for high true SNR. For the example from above, the mean-squared SNR dB error for the case of a true SNR of 1 and an estimated

SNR of 0.7 is 2.399, whereas the error for the case of a true SNR of 100 and an estimated SNR of 97 is 0.017.

7.4.1.3. Mean-Squared Minimum Distance Symbol-Timing Estimation Error. In order to quantify the performance of each of the algorithms with respect to symbol-timing error, it is tempting to consider a simple mean-squared error measure between the true and estimated symbol-timing error, which is given below:

$$\xi_\varepsilon = \frac{1}{N_t} \sum_{n=0}^{N_t-1} \left| \varepsilon - \hat{\varepsilon}^{(n)} \right|^2 \tag{7-9}$$

The problem with using the metric given in Eq. (7-9) is that both symbol-timing errors are assumed to be in the interval $[0, 1)$. However, in reality, each symbol-timing error can be shifted by any integer amount without loss of generality. For example, if the estimated jitter is $\hat{\varepsilon}^{(n)} = 0.75$, this is also equivalent to $\hat{\varepsilon}^{(n)} = \cdots, -1.25, -0.25, 0.75, 1.75, 2.75, \cdots$. This shifting property can cause the metric given in Eq. (7-9) to be overly pessimistic in certain cases.

To see this, consider the case where the true symbol-timing error is $\varepsilon = 0.1$ and the estimated value is $\hat{\varepsilon}^{(n)} = 0.9$. Using Eq. (7-9), we find that $\xi_\varepsilon = 0.64$. However, this error is overly pessimistic, since there is a shifted version of the estimated symbol-timing error (namely $\hat{\varepsilon}^{(n)} = -0.1$) that is closer to the true value of $\varepsilon = 0.1$. This is illustrated in Fig. 7-6. Using this shifted value of $\hat{\varepsilon}^{(n)}$, we obtain $\xi_\varepsilon = 0.04$, which is a more appropriate value for the error between ε and $\hat{\varepsilon}^{(n)}$ in this case.

Thus, a more appropriate measure of the jitter estimation error is to find the *minimum distance* between the true and estimated jitters as the jitters vary over all possible shifted values. Equivalently, we can fix the true jitter to be in the interval $[0, 1)$ and find the shifted version of the estimated jitter that is closest to the true jitter. In other words, a more appropriate measure of the jitter estimation error is to replace each term of the summation in Eq. (7-9) with a term of the form

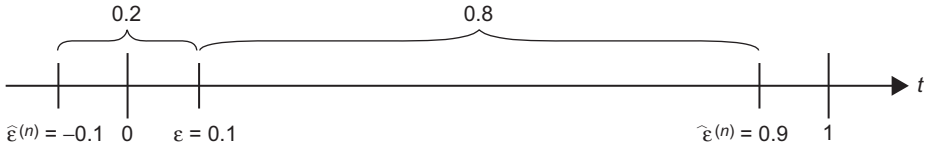

Fig. 7-6. Example of a pessimistic value for the jitter estimation error ξ_ε from Eq. (7.9).

$$\min_{\ell \in \mathbb{Z}} \left\{ \left| \varepsilon - \left(\ell + \hat{\varepsilon}^{\,(n)} \right) \right|^2 \right\} \tag{7-10}$$

where we assume $\varepsilon, \hat{\varepsilon}^{\,(n)} \in [0, 1)$. Fortunately, under the assumption that $\varepsilon, \hat{\varepsilon}^{\,(n)} \in [0, 1)$, we need not look over all values of $\ell \in \mathbb{Z}$ in Eq. (7-10). In particular, we need only look for the minimum value over $\ell = -1, 0, 1$. To see this, note that we have

$$0 \leq \varepsilon < 1, \quad 0 \leq \hat{\varepsilon}^{\,(n)} < 1$$

from which we conclude

$$-1 < \varepsilon - \hat{\varepsilon}^{\,(n)} < 1$$

By adding $-\ell$ to all sides of the inequality, we have

$$-\ell - 1 < \varepsilon - \left(\ell + \hat{\varepsilon}^{\,(n)} \right) < -\ell + 1$$

Now, for $|\ell| \geq 2$, it can be shown that

$$\left| \varepsilon - \left(\ell + \hat{\varepsilon}^{\,(n)} \right) \right|^2 > 1 > \left| \varepsilon - \hat{\varepsilon}^{\,(n)} \right|^2$$

and so the term corresponding to $\ell = 0$ always has a smaller magnitude than those corresponding to $|\ell| \geq 2$. Hence the terms corresponding to $|\ell| \geq 2$ can be ignored in the expression of Eq. (7-10), leaving only $\ell = -1, 0, 1$.

Thus, to ascertain the performance of the algorithms with respect to symbol-timing error, we opted to use the following mean-squared *minimum distance* symbol-timing estimation error:

$$\xi_\varepsilon \triangleq \frac{1}{N_t} \sum_{n=0}^{N_t - 1} \min_{\ell = -1, 0, 1} \left\{ \left| \varepsilon - \left(\ell + \hat{\varepsilon}^{\,(n)} \right) \right|^2 \right\} \tag{7-11}$$

The minimization in each term of Eq. (7-11) ensures that we choose the closest (left, neutral, or right-shifted) estimated jitter to the true one.

We now proceed to present simulation results for the SSME-based data rate estimation algorithms of Sections 7.3.2 and 7.3.3.

7.4.2 Behavior of the SSME-Based Data Rate Estimation Algorithms as a Function of SNR

For all of the simulations considered here, from Eq. (7-1), the data constellation d_k used was quadrature phase-shift keying (QPSK) [1], and the residual frequency offset ω_r was set to zero. To test the data rate estimation algorithms of Sections 7.3.2 and 7.3.3, we opted to choose the following input parameters:

$$\begin{aligned} B &= 2 \\ \ell_{\max} &= 3 \\ N_{\hat{\varepsilon},b} &= 2 \\ N_{\text{obs}} &= 64 \\ N_b &= 16 \end{aligned}$$

It should be noted that the choice of N_{obs} and N_b here implies that we have an observation time epoch equal to $N_{\text{obs}}N_b = 1{,}024$ *lowest rate* symbols. This time epoch was fixed here for all possible data rates in order to reflect the fact that we are assumed to have no a priori knowledge of the data rate. As such, this intuitively implies that on average the SSME will be able to estimate the SNR more accurately for higher data rates. The reason for this is that, for a fixed time epoch, the SSME will have more observations the higher the data rate becomes. This will result not only in an increase in the accuracy of the SNR estimate for higher data rates, but also often in a better probability of misclassification and jitter estimation error, as will soon be shown.

For a preliminary set of simulations, suppose that the symbol-timing error is zero (i.e., $\varepsilon = 0$), and the true SNR R is varied from -10 dB to -3 dB.[3] In Fig.7-7, we have plotted the probability of misclassification, P_m from Eq. (7-7), as a function of SNR using (a) the algorithm of Section 7.3.2 and (b) the algorithm of Section 7.3.3. As can be seen, the algorithm of Section 7.3.2 outperforms that of Section 7.3.3 for the higher data rates, but fails to do so for the lower ones. The reason for this is that the algorithm of Section 7.3.2 will often prematurely terminate, which is beneficial for higher true data rates and detrimental for lower true ones.

One unusual phenomenon that can be observed from Fig. 7-7 is that the curves cross for different values of the true data rate. This appears counterintuitive, since we should expect the higher data rates to be classified correctly more often than the lower data rates (as there are a larger number of observations in

[3] The reason for varying the true SNR over such low values is to reflect the fact that, in the DSN, the SNR is typically rather small.

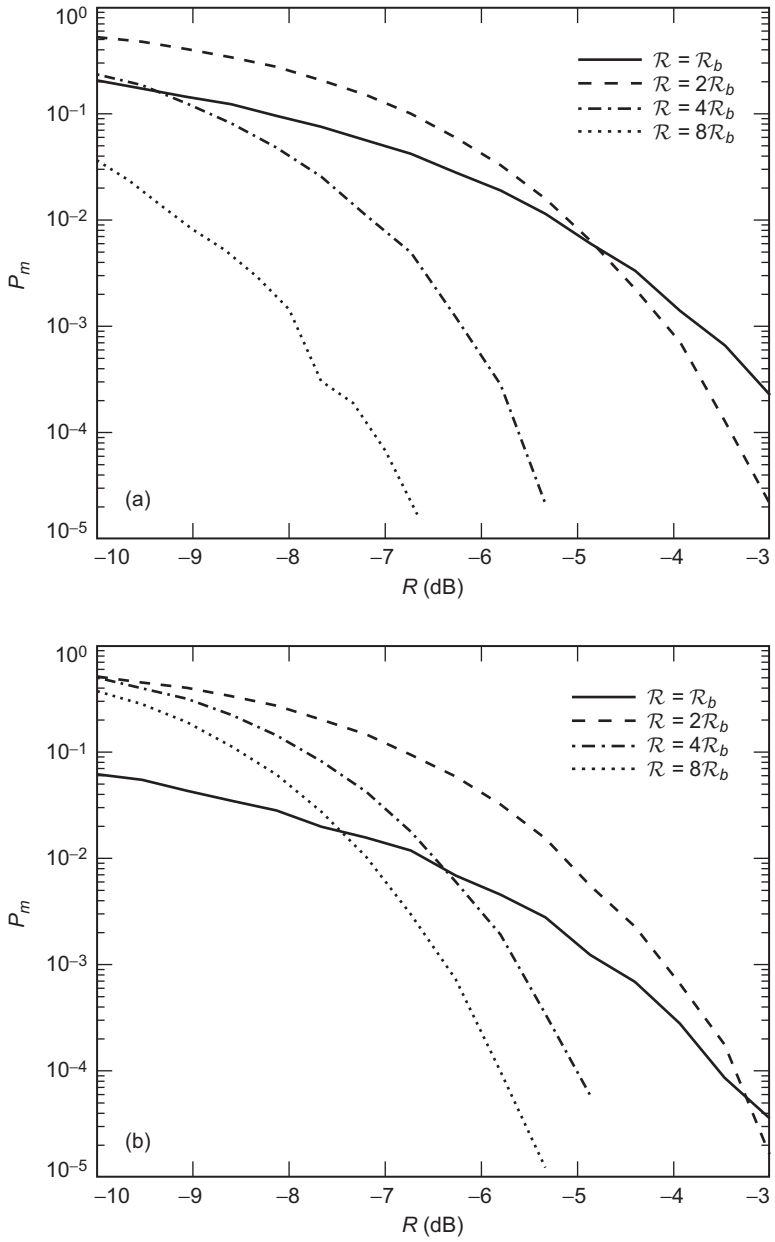

Fig. 7-7. Probability of data rate misclassification as a function of SNR using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3.

these cases). However, when the true SNR is low, the factor corresponding to the number of observations in the expression for the mean of the SSME SNR estimate becomes non-negligible (see the equation above Eq. (7-3) for more details). This most likely is the reason that the curves cross at lower true SNR. At higher true SNR, the mean of the SNR estimate becomes less sensitive to the number of observations and so we expect the higher rates to be classified correctly more often than the lower rates. This is indeed the case here, as can be seen in Fig. 7-7 when the true SNR is near -3 dB.

In order to accurately compare the two algorithms, one figure of merit that can be used is the *average* probability of misclassification, which we denote here by $\overline{P}_m$. If p_ℓ denotes the probability that the true data rate is $\mathcal{R} = B^\ell \mathcal{R}_b$ and $P_{m|\ell}$ denotes the probability of misclassification given that the true data rate is $B^\ell \mathcal{R}_b$, then by the theorem of total probability [7], we have

$$\overline{P}_m = \sum_{\ell=0}^{\ell_{\max}} p_\ell P_{m|\ell} \tag{7-12}$$

Assuming that the true data rates are equiprobable (i.e., $p_\ell = 1/(\ell_{\max}+1)$ for all ℓ), Eq. (7-12) becomes

$$\overline{P}_m = \frac{1}{\ell_{\max}+1} \sum_{\ell=0}^{\ell_{\max}} P_{m|\ell}$$

A plot of $\overline{P}_m$ as a function of the true SNR R is shown in Fig. 7-8 for equiprobable data rates. From this, it can be seen that, for lower SNR, the algorithm of Section 7.3.2 yields a better average probability of misclassification, whereas for higher SNR (above about -7.3 dB), the algorithm of Section 7.3.3 performs better. Since the desired SNR for a DSN-type application is -6 dB or greater (in order to achieve good performance for the turbo codes expected to be used), this implies that the GLRT-type algorithm of Section 7.3.3 is best suited here.

To further compare the two algorithms, in Fig. 7-9 we have plotted the observed mean-squared SNR dB estimation error, ξ_R from Eq. (7-8), for the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3. As can be seen, the estimation error always decreased monotonically with SNR for each data rate. Furthermore, it can be seen that the error decreased almost geometrically as the data rate increased. These two phenomena are consistent with the fact that the SSME yields a better estimate of the SNR as both the true SNR and number of observations increase.

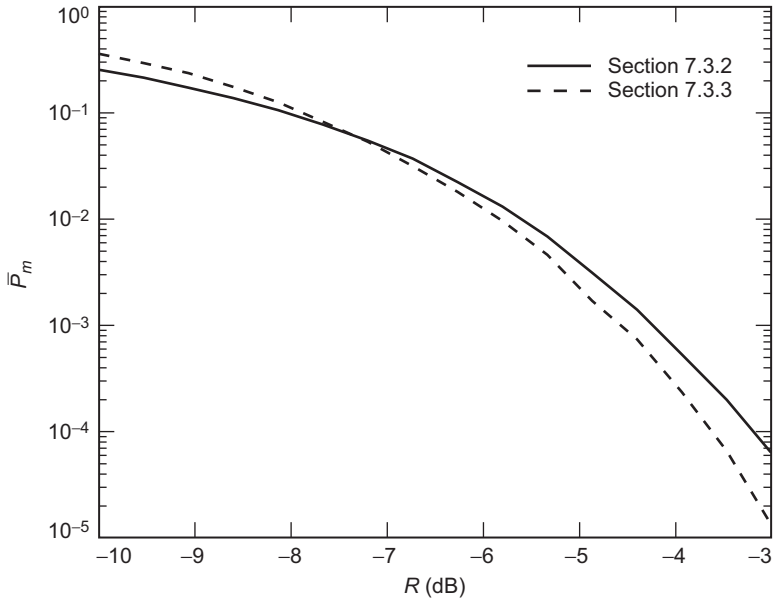

Fig. 7-8. Average probability of misclassification as a function of SNR for the algorithms of Sections 7.3.2 and 7.3.3.

Analogous to the average probability of misclassification $\overline{P}_m$ given in Eq. (7-12), we can quantitatively compare both algorithms in terms of the average mean-squared SNR dB estimation error $\overline{\xi}_R$ given by

$$\overline{\xi}_R = \sum_{\ell=0}^{\ell_{\max}} p_\ell \xi_{R|\ell} \tag{7-13}$$

where $\xi_{R|\ell}$ is the mean-squared SNR dB error given that the true data rate is $\mathcal{R} = B^\ell \mathcal{R}_b$. Assuming equiprobable data rates in Eq. (7-13), a plot of $\overline{\xi}_R$ as a function of the true SNR R is shown in Fig.7-10 for both algorithms. As can be seen, the GLRT-type algorithm of Section 7.3.3 always outperformed that of Section 7.3.2, although for larger SNR (near -3 dB), the two performed nearly identically. This is consistent with the intuition that the two algorithms should be performing increasingly similarly as the true SNR increases since the SNR estimates are more accurate in this case.

As a final measure of comparison between the two algorithms, the observed mean-squared minimum distance symbol-timing estimation error, ξ_ε from Eq. (7-11), is shown in Fig.7-11. From this, it can be seen that the algorithm of Section 7.3.2 yielded a good estimate for the higher data rates but suffered for

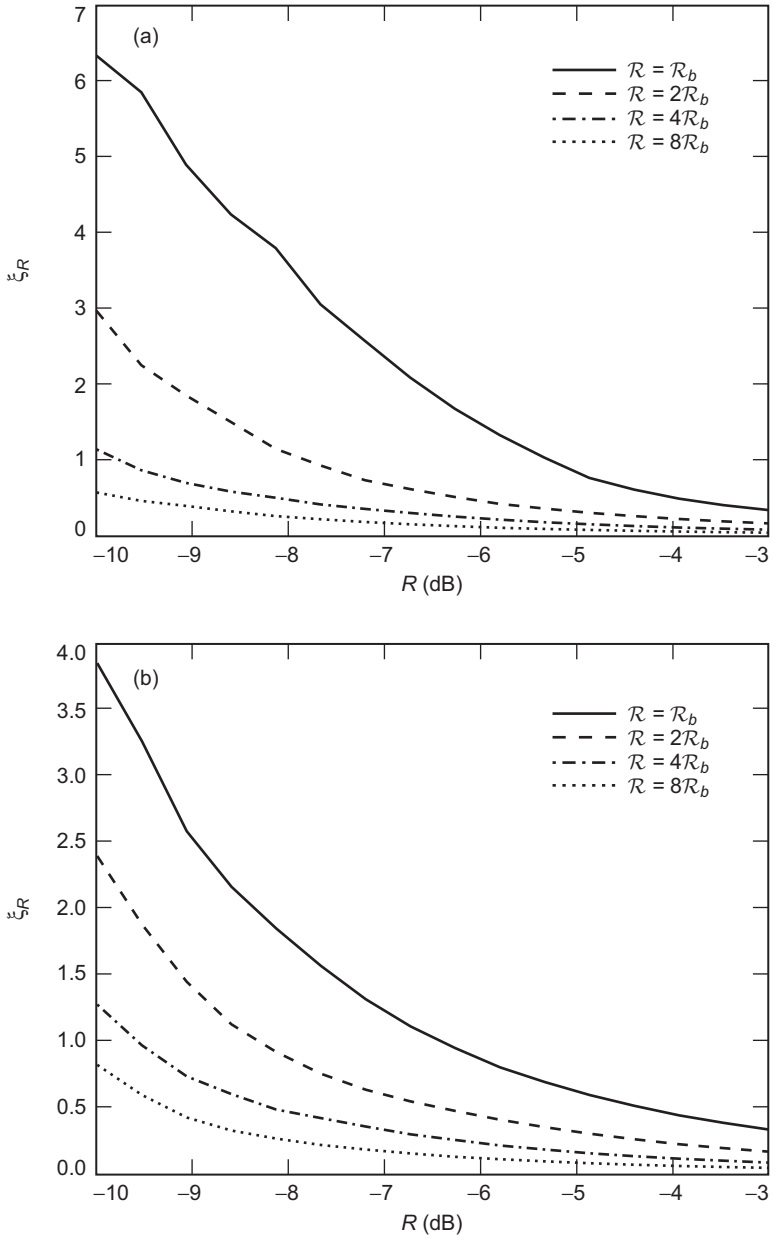

Fig. 7-9. Mean-squared SNR decibel estimation error as a function of the true SNR using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3.

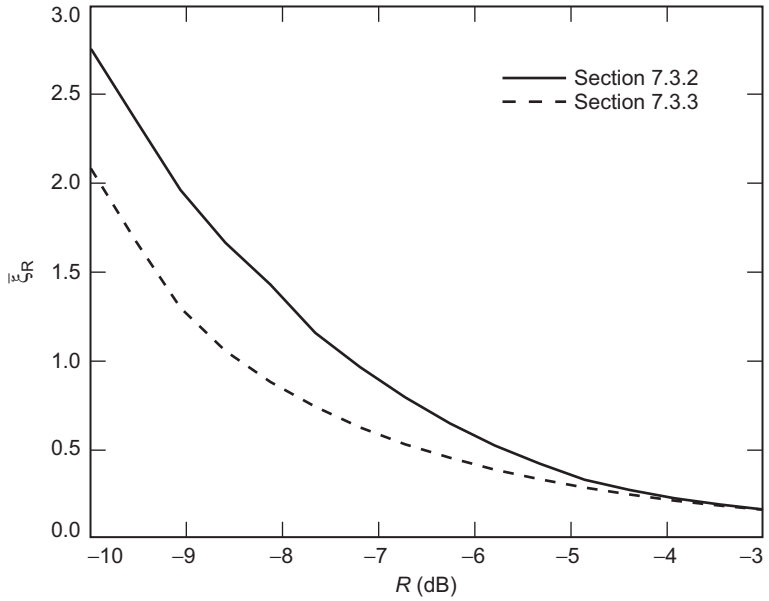

Fig. 7-10. Average mean-squared SNR decibel estimation error as a function of the true SNR for the algorithms of Sections 7.3.2 and 7.3.3.

the lower ones. This perhaps is due to the inherent premature halting possibility of the algorithm, as discussed earlier. For the algorithm of Section 7.3.3, it can be seen that at low SNR the error is large for all rates and that, with the exception of the lowest data rate, for a fixed SNR the error decreased as the rate increased.

As before, to quantitatively compare both algorithms, we can do so by computing the average mean-squared minimum distance symbol-timing estimation error $\overline{\xi}_\varepsilon$ given by

$$\overline{\xi}_\varepsilon = \sum_{\ell=0}^{\ell_{\max}} p_\ell \xi_{\varepsilon|\ell} \tag{7-14}$$

where $\xi_{\varepsilon|\ell}$ denotes the symbol-timing estimation error given that the true data rate is $\mathcal{R} = B^\ell \mathcal{R}_b$. Assuming equiprobable data rates in Eq. (7-14), a plot of $\overline{\xi}_\varepsilon$ as a function of the SNR R is shown in Fig. 7-12. From this, it can be seen that for low SNR the algorithm of Section 7.3.2 notably outperformed the algorithm of Section 7.3.3. Above about -7.1 dB, however, the opposite scenario took place. As the desired mode of operation for the autonomous radio is above -6 dB, this implies that once again the GLRT-type algorithm

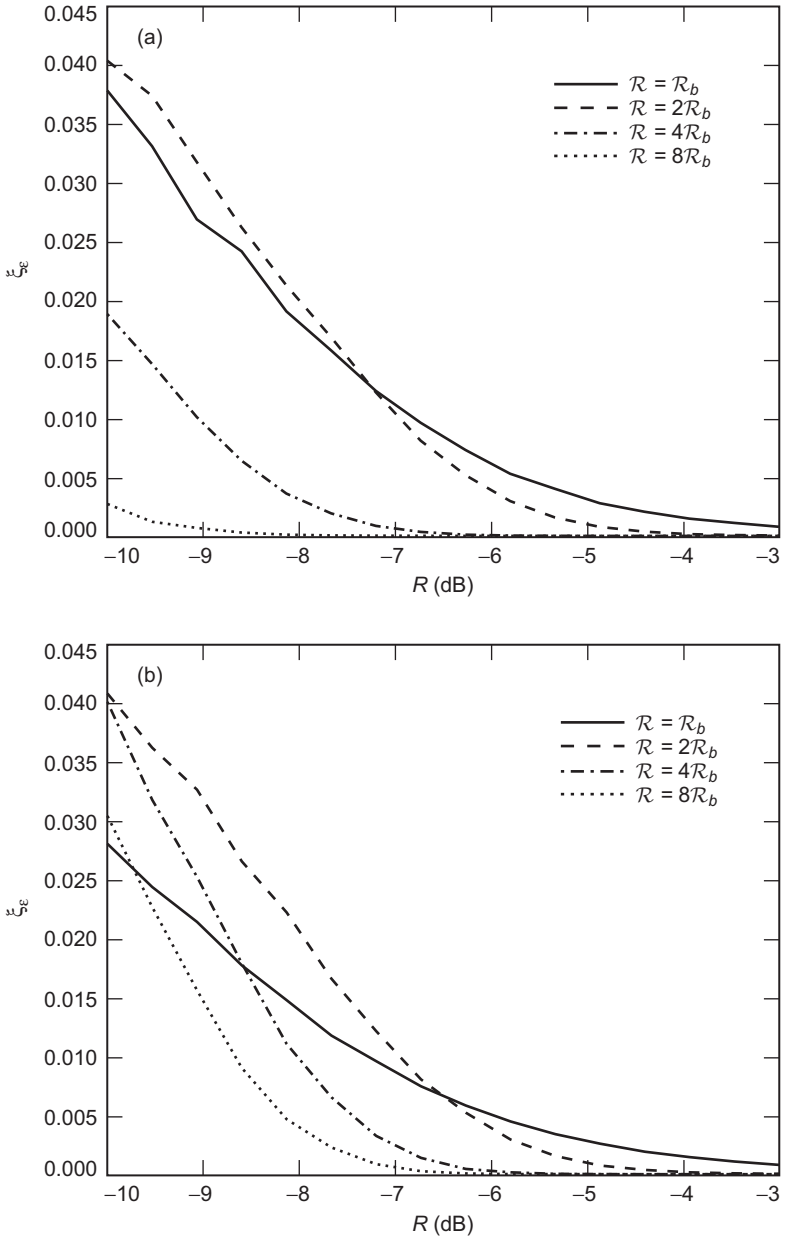

Fig. 7-11. Mean-squared minimum distance symbol timing estimation error as a function of SNR using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3.

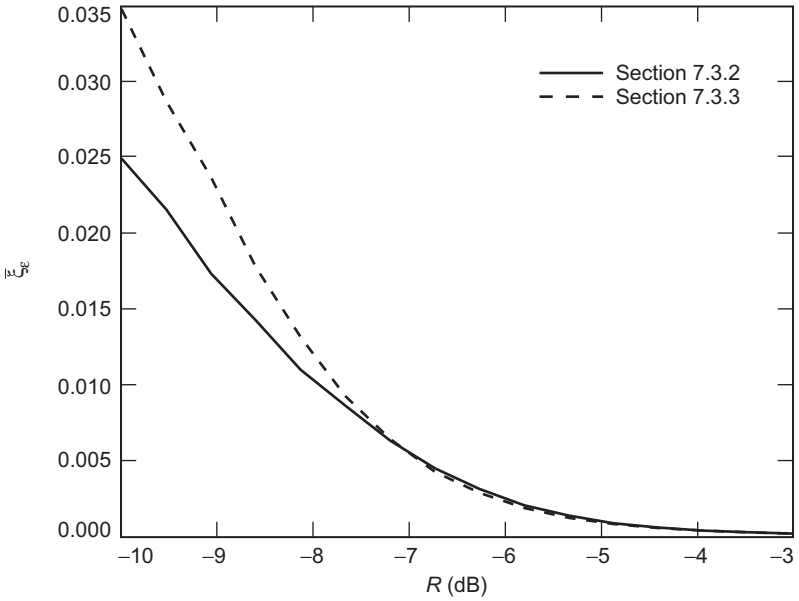

Fig. 7-12. Average mean-squared minimum distance symbol-timing estimation error as a function of SNR for the algorithms of Sections 7.3.2 and 7.3.3.

of Section 7.3.3 is best suited here. It should be noted, however, that these algorithms can be used to obtain only a coarse estimate of the symbol-timing error and that once the data rate has been successfully classified, a finer estimate of the jitter can be obtained through the use of a DTTL [5].

7.4.3 Behavior of the SSME-Based Data Rate Estimation Algorithms as a Function of Symbol-Timing Error

In the previous section, we considered the performance of the data rate estimation algorithms of Sections 7.3.2 and 7.3.3 for a varying SNR and a fixed symbol-timing error. Here, we investigate the performance of the algorithms as a function of the jitter for fixed SNR. As the target SNR for the autonomous radio for the DSN is above -6 dB (in order to achieve good performance from the turbo codes to be used for error correction), the SNR here was fixed at -6 dB.

To illustrate the effects of quantizing and coarsely estimating the symbol-timing error on estimating the data rate, suppose that the true data rate is $\mathcal{R} = 2\mathcal{R}_b$. Plots of the observed probability of misclassification are shown in Fig. 7-13 for (a) the algorithm of Section 7.3.2 and (b) the algorithm of Section 7.3.3. As can be seen, for both methods the probability appears to oscillate back and forth as the jitter varies. It can be seen that both plots

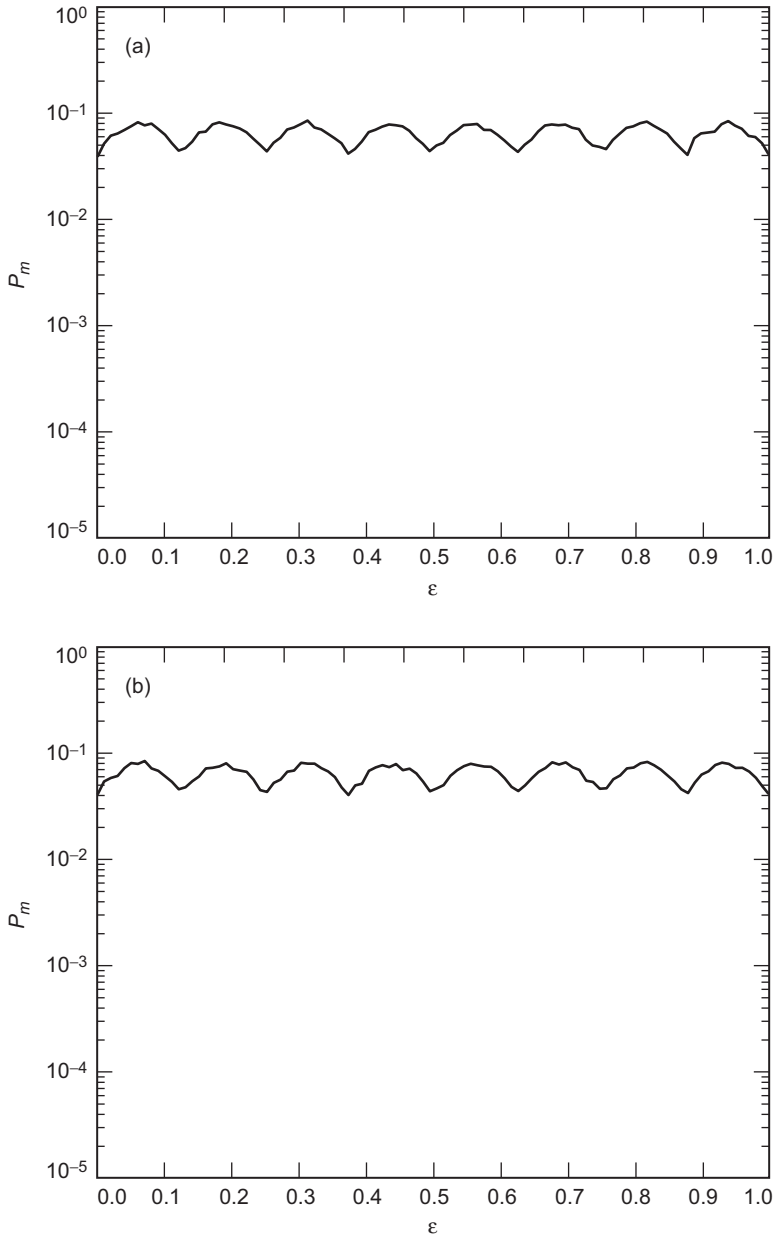

Fig. 7-13. Probability of data rate misclassification as a function of the jitter using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3. (The true data rate is $\mathcal{R} = 2\mathcal{R}_b$.)

appear to have eight equispaced local maxima. The reason for this is due to the quantization of the symbol-timing error. Recall that from Section 7.4.2 the basic number of quantization levels was $N_{\hat{\varepsilon},b} = 2$. This implies that at the true data rate $\mathcal{R} = 2\mathcal{R}_b = 2^1\mathcal{R}_b$ the symbol-timing error is quantized to $N_{\hat{\varepsilon},s} = 2^{\ell_{\max}-1}N_{\hat{\varepsilon},b} = 2^2 N_{\hat{\varepsilon},b} = 8$ steps by using Eq. (7-5). These steps are equispaced about the interval $[0, 1)$ and are of the form $n/8$ for $0 \leq n \leq 7$. Every time each of the data rate estimation algorithms is run, each method chooses the quantized value of the jitter that is the "best fit" in some sense to the true jitter. As the true jitter itself is varied, it is evident that there will be *ambiguous* values of the symbol-timing error that occur directly in between the quantized values. This is illustrated in Fig.7-14 for the case of 8 quantization steps here. From Fig. 7-13, it is clear that the probability of misclassification becomes locally maximal almost precisely at these ambiguous jitter value locations.

To further observe the effects of varying the symbol-timing error, a plot of the observed mean-squared SNR decibel error is shown in Fig. 7-15 for both algorithms. Note that, unlike the probability of misclassification, for both methods the error in estimating the SNR remains approximately constant as the jitter is varied. The reason for this robustness most likely comes from the fact that, with a sufficient number of quantization steps, the "best fit" jitter value to the true one chosen for the SSME will incur only a small degradation in the mean of the SNR estimate. See Chapter 6 for more details on the quantitative amount of this degradation.

As a final measure of the effects of varying symbol-timing error on the data rate estimation algorithms of Sections 7.3.2 and 7.3.3, a plot of the observed mean-squared minimum distance jitter estimation error for each algorithm is shown in Fig. 7-16. Like the probability of misclassification plots of Fig. 7-13, it can be seen that the error for both algorithms oscillates back and forth as the jitter varies. Also as before, each plot appears to have eight equispaced local maxima that occur approximately at the locations corresponding to the ambiguous values of the symbol-timing error. This observation is consistent with

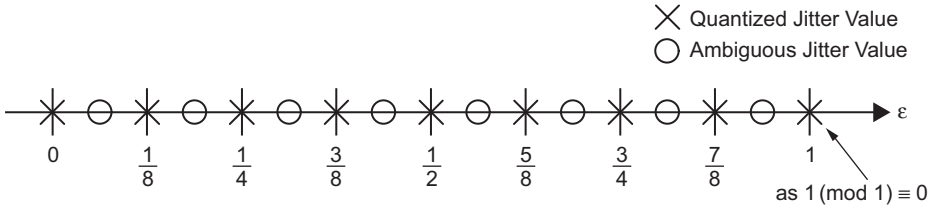

Fig. 7-14. Example showing the quantized symbol-timing error values for $N_{\hat{\varepsilon},s} = 8$ along with the *ambiguous* jitter values.

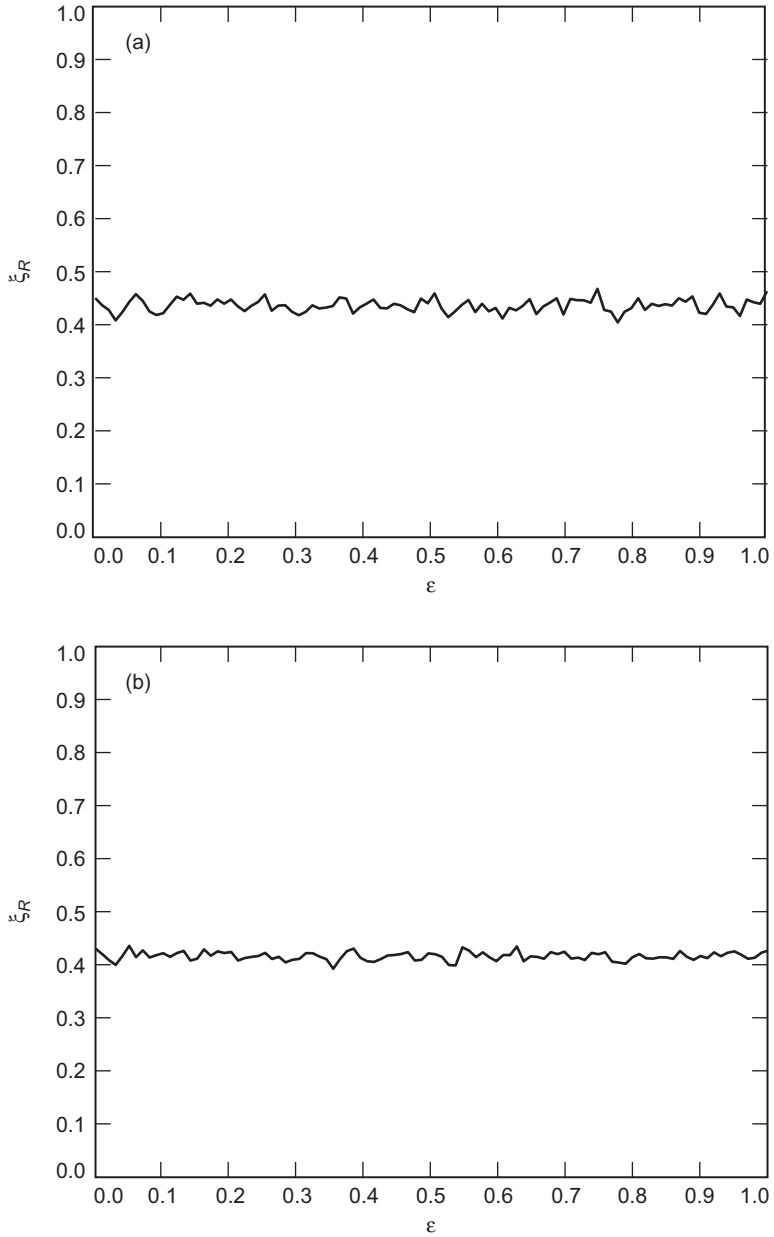

Fig. 7-15. Mean-squared SNR decibel estimation error as a function of the jitter using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3. (The true data rate is $\mathcal{R} = 2\mathcal{R}_b$.)

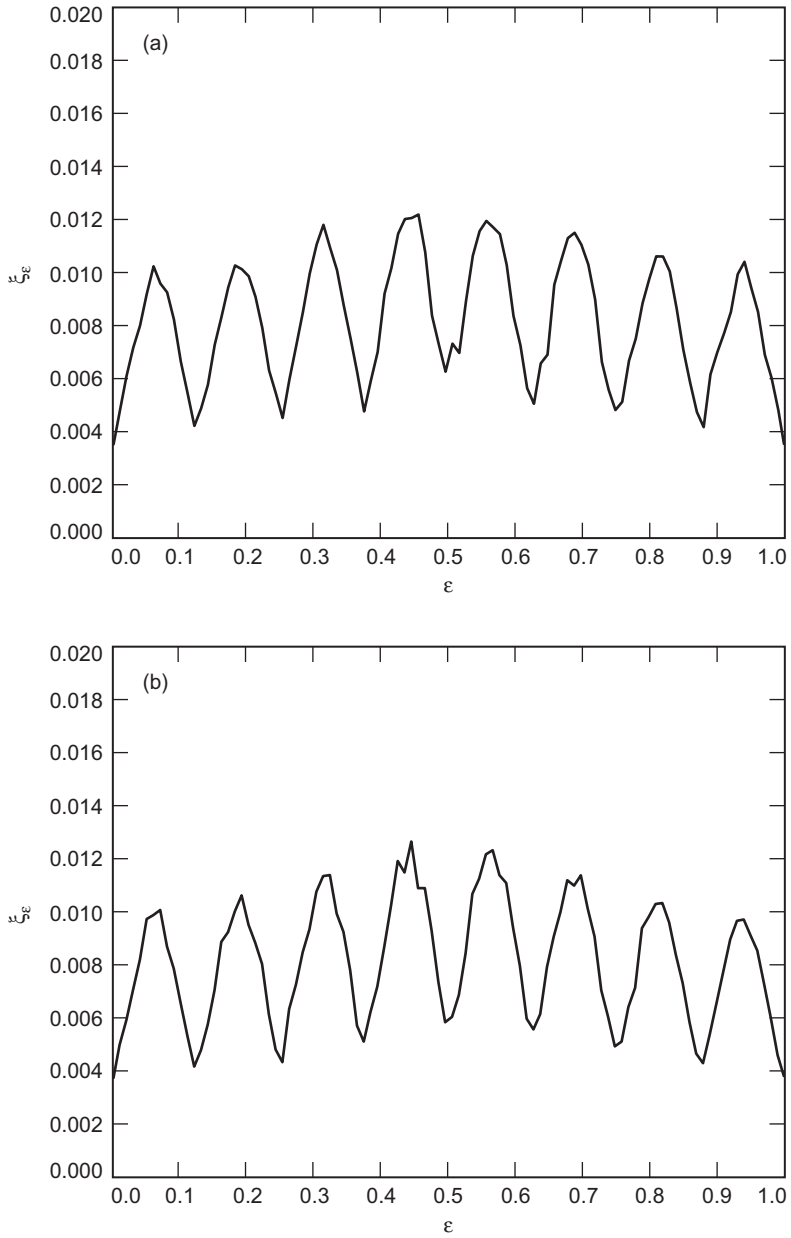

Fig. 7-16. The mean-squared minimum distance symbol-timing estimation error as a function of the jitter using the algorithms of (a) Section 7.3.2 and (b) Section 7.3.3. (The true data rate is $\mathcal{R} = 2\mathcal{R}_b$.)

the intuition that the estimation process should suffer the most degradation at the ambiguous jitter values. One new phenomenon that can be observed from the plots of Fig.7-16 is that, for both algorithms, the error appears symmetric about $\varepsilon = 1/2$ and seems to generally increase as $\varepsilon \to 1/2$ from either direction. The reason for this phenomenon is not clear at this point and requires further investigation.

At this point, a few comments are in order. Had the above simulations been run for another true data rate, say at $\mathcal{R} = 4\mathcal{R}_b$, then there would have been 4 ambiguous jitter values instead of 8, since $N_{\hat{\varepsilon},s} = 2^{3-2}N_{\hat{\varepsilon},b} = 4$ in this case. The same observations regarding the performance metrics would still hold true, with the exception that the degradation in performance due to fewer jitter quantization steps would be more pronounced. In general, with a true data rate of $\mathcal{R} = B^{\ell}\mathcal{R}_b$, the number of symbol-timing error quantization steps at the true data rate is $N_{\hat{\varepsilon},s} = B^{\ell_{\max}-\ell}N_{\hat{\varepsilon},b}$ from Eq. (7-5). This suggests that an implicit trade-off in performance exists between the data rate and the granularity of the symbol-timing error. For a fixed observation time epoch of the received signal, the higher the data rate, the more observations we have to help improve the estimate of the mean of the SNR of the SSME. However, at the same time, we also have an increased sensitivity to the symbol-timing error in this case. Conversely, the lower the data rate, the fewer samples there are to estimate the mean of the SNR. However, at the same time, we also have more robustness with respect to the symbol-timing error.

The effect of increasing the basic number of symbol-timing error quantization steps $N_{\hat{\varepsilon},b}$ is to increase the number of ambiguous jitter values but at the same time to decrease the degradation at these values. Thus, the estimation becomes more robust in this case. However, this comes at the price of increased computational complexity, as well as an increase in the oversampling rate of the received signal. For the Electra radio (see [2] and Chapter 2), the sampling rate is 4 times the highest data rate, and so the maximum value of $N_{\hat{\varepsilon},b}$ that can be used for this system is $N_{\hat{\varepsilon},b} = 4$. Although this value may appear to be small, for most applications this should be sufficient for estimating the data rate and SNR reasonably well. As mentioned above, once the data rate has been classified correctly, the symbol-timing error can be finely estimated through the use of a DTTL (see [5] and Chapter 10).

References

[1] M. K. Simon, S. M. Hinedi, and W. C. Lindsey, *Digital Communications Techniques: Signal Design and Detection*, Upper Saddle River, New Jersey: Prentice Hall PTR, 1994.

[2] S. F. Franklin, J. P. Slonski, Jr., S. Kerridge, G. Noreen, S. Townes, E. Schwartzbaum, S. Synnott, M. Deutsch, C. Edwards, A. Devereaux, R. Austin, B. Edwards, J. J. Scozzafava, D. M. Boroson, W. T. Roberts, A. Biswas, A. D. Pillsbury, F. I. Khatri, J. Sharma, and T. Komarek, "The 2009 Mars Telecom Orbiter Mission," *Proceedings of the IEEE Aerospace Conference*, Big Sky, Montana, pp. 437–456, March 2004.

[3] M. Simon and S. Dolinar, "Improving Signal-to-Noise Ratio Estimation for Autonomous Receivers," *The Interplanetary Network Progress Report*, vol. 42-159, Jet Propulsion Laboratory, Pasadena, California, pp. 1–19, November 15, 2004. http://ipnpr/progress_report/42-159/159D.pdf

[4] S. M. Kay, *Fundamentals of Statistical Signal Processing, Volume II: Detection Theory*, Upper Saddle River, New Jersey: Prentice Hall PTR, 1998.

[5] W. C. Lindsey and M. K. Simon, *Telecommunication Systems Engineering*, New York: Dover Publications, 1973.

[6] M. Simon and D. Divsalar, "Data Format Classification for Autonomous Radio Receivers," *The Interplanetary Network Progress Report*, vol. 42-159, Jet Propulsion Laboratory, Pasadena, California, pp. 1–27, November 15, 2004. http://ipnpr/progress_report/42-159/159G.pdf

[7] A. Papoulis and S. U. Pillai, *Probability, Random Variables and Stochastic Processes*, fourth ed., New York: McGraw-Hill, 2002.